UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/516,318	03/01/2000	Charles Paul Siska JR.	60944.1600	3654

500 7590 04/20/2004

SEED INTELLECTUAL PROPERTY LAW GROUP PLLC
701 FIFTH AVE
SUITE 6300
SEATTLE, WA 98104-7092

EXAMINER	
WOOD, WILLIAM H	
ART UNIT	PAPER NUMBER
2124	18

DATE MAILED: 04/20/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/516,318	**Applicant(s)** SISKA, CHARLES PAUL
	Examiner William H. Wood	**Art Unit** 2124

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 February 2004*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *15*.

4) □ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

Claims 1-20 are pending and have been examined.

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 4 February 2004 has been entered.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. Claims 1-20 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter, which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. Applicant's amended claims recite "ADL representation generates a representation of structure". Applicant cited pages 3 and 4 of the specification as giving support for this new limitation. The cited portions have failed to provide convincing evidence that Applicant intended the ADL of the invention to "generate a representation of structure". Page 3, lines 6-8, Applicant generally indicates

nML, LISA, ISDL and RADL as common ADLs. Page 9, lines 16-19, Applicant specifically states nML, LISA and ISDL are acceptable for implementing the ADL of the current invention. ISDL and nML are generally not considered ADLs incorporating structure representations (at most they do so limitedly). Additionally, Applicant provided the document **Qin** et al., “Architecture Description Languages for Retargetable Compilation” as a non-prior art reference in support of the universal fact that ISDL, nML and SLED are not a representation of structure ADLs (see table 1 on page 20). Therefore, as Applicant disclosed of an invention implemented using nML and ISDL (non structural representation), Applicant cannot have had possession of an invention, which uses ADLs of structural representations. The cited passage only discusses ADLs in general and does not specify a structural representation ADL or some other type of ADL. Evidence that claims 1 fail(s) to correspond in scope with that which applicant(s) regard as the invention can be found in Paper No. 17 filed 04 February 2004. In that paper, applicant has stated “One having ordinary skill in the art will appreciate that a shorthand notation for the quoted material is that the applicant's ADL is used to produce a ‘representation of structure’” (page 10, line 23) and “Specifically, the applicant directs the Examiner's attention to Table 1 of *Architecture Description Languages for Retargetable Compilation*, W. Qui and S Malik …, which clearly shows that SLED does not generate a representation of structure” (page 10, last sentence to page 11, table 1), and these statements indicate that the invention is different from what is defined in the claim(s) because the cited table 1 also demonstrates nML and ISDL (which are implementations of Applicant's invention) are not ADLs with representations of structure. Thus, the current claims cannot be regarded by Applicant as reflective of the current invention.

4. Claims 1-20 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Amended independent claims 1, 4, 9 and 12... contain the new limitation "ADL representation generates a representation of structure". It is unclear from the cited portions of the Specification (pages 3 and 4) as to the interpretation of this phrase. The intent of the limitation could be the ADL generates a structure (an assembler or debugger for example) or the ADL defines (is) a structural representation (an abstract encoding defining a processor). The Specification appears to support both interpretations. In the rejections below, "ADL representation generating a representation of structure" is interpreted as "ADL describes/defines/is a representation of structure".

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 1-15 and 19-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ramsey et al., "Specifying Representations of Machine Instructions" in view of Gupta et al. (USPN 6,385,757) in further view of Russo et al, "Automatic Architecture Evaluates For Hardware/Software Codesign".

In regard to claim 1, Ramsey disclosed the limitations:

> i) *method for producing code in an architecture description language* (page 496, bracket 3)
>
> ii) *reading an opcode summary table* (page 496, bracket A; page 497, bracket 5 indicates the opcode tables being read for information)
>
> iii) *analyzing said opcode summary table to determine the layout of said opcode summary table* (page 495, bracket B illustrates differing types of instructions which would need to be analyzed in the table in order to be effectively implemented; page 495, bracket 1 indicates instructions differ and therefore to be properly handled the opcode table would need to be analyzed; page 497, bracket 5 indicates organizing the tables in a hierarchy for analysis purposes; page 499, bracket 9 indicates groups)
>
> iv) *generating code for an instruction in architecture description language format* (page 495, section 2)
>
> v) *repeating said generating step for each line on said opcode summary table* (necessary in order to build a complete machine description)
>
> vi) *resulting in an architecture description language (ADL) representation of the opcode summary table (page 496, brackets 3, 4 and A; page 497, brackets 5 and 6; page 495, first paragraph)*

Ramsey did not explicitly state the method being computerized or in other words automated without a programmer. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state ADL representation generates a representation of structure (as interpreted to mean: ADL representation generates a definition of the microprocessor structure). Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph). Russo demonstrated that it was known at the time of invention utilize differing ADLs for differing design purposes including Mimola a structural representation ADL (pages 47-48, section 2). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as representation of structure) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (Russo: pages 47-48, section 2.1; Mimola is useful for synthesizing hardware) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

In regard to claim 2, neither Ramsey nor Gupta explicitly stated the limitation *where the opcode summary table is provided in a spreadsheet*. Official Notice is taken that it was known at the time of invention to use a spreadsheet to represent a table. It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's and Gupta's combined system of ADL formation with a spreadsheet table. This implementation would have been obvious because one of ordinary skill in the art would be motivated to utilize a highly flexible method of maintaining and changing the instruction set for differing architectures.

In regard to claim 3, neither Ramsey nor Gupta explicitly stated the limitation *where the opcode summary table is provided in a comma separated value format*. Official Notice is taken that it was known at the time of invention to use comma separated value format to represent a table. It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's and Gupta's combined system of ADL formation with a comma separated value format table. This implementation would have been obvious because one of ordinary skill in the art would be motivated to utilize a highly flexible method of maintaining and changing the instruction set for differing architectures.

In regard to claim 4, Ramsey disclosed the limitations:

i) *method of producing code in an architecture description language format* (page 496, bracket 3)

ii) *reading an opcode summary table* (page 496, bracket A; page 497, bracket 5 indicates the opcode table being composed of several tables, but not any less a table)

iii) *analyzing said opcode summary table to determine the layout of said opcode summary table* (page 495, bracket B illustrates differing types of instructions which would need to be analyzed in the table in order to be effectively implemented; page 495, bracket 1 indicates instructions differ and therefore to be properly handled the opcode table would need to be analyzed; page 497, bracket 5 indicates organizing the tables in a hierarchy for analysis purposes)

iv) *determining the beginning of a group from said opcode summary table* (page 497, bracket 5; page 499, bracket 9)

v) *generating root code for the hierarchy in architecture description language format based on said grouping* (page 499, bracket 9; implicit patterns; page 497, bracket 6),

vi) *cycling through each group to generate detailed code in architecture language format* (necessary in order to build a complete machine description)

vii) *repeating said cycling step until the end of the opcode summary table is reached* (necessary in order to build a complete machine description)

Ramsey did not explicitly state the method being computerized or in other words automated without a programmer. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state ADL representation generates a representation of structure (as interpreted to mean: ADL representation generates a definition of the microprocessor structure). Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph). Russo demonstrated that it was known at the time of invention utilize differing ADLs for differing design purposes including Mimola a structural representation ADL (pages 47-48, section 2). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as representation of structure) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (Russo: pages 47-48, section 2.1; Mimola is useful for synthesizing hardware) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

In regard to claims 5 and 6, the claims are corresponding to claims 2 and 3, respectively, and only differing in the claim to which they depend. The independent claims have been rejected in the same manner and therefore claims 5 and 6 are rejected the same way as claims 2 and 3 above.

In regard to claim 7, Ramsey disclosed the limitation *where the opcode summary table is pre-formatted such that the opcodes are separated into groups prior to being read* (page 499, bracket 9).

In regard to claim 8, Ramsey and Gupta did not explicitly state the limitation *where said cycle step further comprises determining the presence of sub-groups within said group and generating detailed code for each sub-group within said group.* This step is implied, however, by Ramsey by page 497, bracket 5's mention of hierarchy of tables and page

499, bracket 9's groupings and implicit patterns. It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey with finding any similarities and thus groups and therefore subgroups within a hierarchy (groups being disclosed on page 499, bracket 9). This implementation would have been obvious because one of ordinary skill in the art would be motivated to make the most use of grouping implicit patterns and hierarchies of instructions, in order to logically process in as efficient manner as possible a large group of instructions/opcodes. Logically defining the instructions as such provides for an easy way to maintain the tables of opcodes.

In regard to claim 9, Ramsey disclosed the limitations:

> i) *a first computer code section for reading an opcode summary table having a plurality of entries representative of a like plurality of microprocessor instructions* (page 496, bracket 3)
>
> ii) *a second computer code section for producing a group of at least two of said entries in accordance with a grouping criteria* (page 497, bracket 5; page 499, bracket 9)
>
> iii) *a third computer code section for generating an encoded representation of said grouping* (page 499, bracket 9)

Ramsey did not explicitly state the above steps being taken by a computer program. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state ADL representation generates a representation of structure (as interpreted to mean: ADL representation generates a definition of the microprocessor structure). Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph). Russo demonstrated that it was known at the time of invention utilize differing ADLs for differing design purposes including Mimola a structural representation ADL (pages 47-48, section 2). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as representation of structure) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (Russo: pages 47-48, section 2.1; Mimola is useful for synthesizing hardware) and the benefits of opcode table features of SLED (Ramsey:

page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

In regard to claim 10, Ramsey demonstrated:

- *reading an opcode summary table* (page 496, bracket A; page 497, bracket 5 indicates the opcode tables being read for information)*;*
- *analyzing said opcode summary table to determine the layout of said opcode summary table and constructing an opcode super group based on at least two opcode groups identified by said analyzing* (page 495, bracket B illustrates differing types of instructions which would need to be analyzed in the table in order to be effectively implemented; page 495, bracket 1 indicates instructions differ and therefore to be properly handled the opcode table would need to be analyzed; page 497, bracket 5 indicates organizing the tables in a hierarchy for analysis purposes; page 499, bracket 9 indicates groups)*; and*
- *generating code for an instruction in architecture description language format based on said opcode super group* (Ramsey by page 497, bracket 5's mention of hierarchy of tables and page 499, bracket 9's groupings and implicit patterns).

Ramsey did not explicitly state the above steps being taken by a computer program. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state ADL representation generates a representation of structure (as interpreted to mean: ADL representation generates a definition of the microprocessor structure). Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph). Russo demonstrated that it was known at the time of invention utilize differing ADLs for differing design purposes including Mimola a structural representation ADL (pages 47-48, section 2). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as representation of structure) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (Russo: pages 47-48, section 2.1; Mimola is useful for synthesizing hardware) and the benefits of opcode table features of SLED (Ramsey:

page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

In regard to claim 11, Ramsey disclosed the limitations:

Hanono did not explicitly state:

- *reading an opcode super group table* (page 496, bracket A; page 497, bracket 5 indicates the opcode tables being read for information);
- *analyzing said opcode super group table to determine a layout of said opcode super group table* (page 497, bracket 5);
- *determining a presence of a sub-group from said opcode super group table* (page 497, bracket 5);
- *generating root code in architecture description language format based on the sub-group* (Ramsey by page 497, bracket 5's mention of hierarchy of tables and page 499, bracket 9's groupings and implicit patterns);
- *cycling to generate detailed code for the sub-group in architecture description language format* (necessary in order to build a complete machine description);
- *repeating said cycling and determining until the end of the opcode super group table is reached* (necessary in order to build a complete machine description)

Ramsey did not explicitly state the above steps being taken by a computer program. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state ADL representation generates a representation of structure (as interpreted to mean: ADL representation generates a definition of the microprocessor structure). Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph). Russo demonstrated that it was known at the time of invention utilize differing ADLs for differing design purposes including Mimola a structural representation ADL (pages 47-48, section 2). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as representation of structure) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (Russo: pages 47-48, section 2.1; Mimola is useful for synthesizing hardware) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

In regard to claim 12, Ramsey, Gupta and Russo disclosed the limitations of the claim as noted above under claims 10 and 11. Claim 12 is a program for performing the computerized methods disclosed and possesses limitations as such.

Claim 13

Ramsey, Gupta and Russo disclosed the method of claim 1, wherein the representation of structure comprises an ADL representation of a microprocessor (Russo: pages 47-48, section 2).

Claim 14

Ramsey, Gupta and Russo disclosed the method of claim 1, wherein the representation of structure comprises an ADL microprocessor representation utilizable by an assembler generator (Ramsey: page 493, first paragraph).

Claim 15

Ramsey, Gupta and Russo disclosed the method of claim 1, wherein the representation of structure comprises an ADL microprocessor representation utilizable by a simulator generator (Ramsey: page 520, third line).

Claims 19

The claim 19 corresponds to claim 1 and is rejected by Ramsey, Gupta and Russo in the same manner.

Claim 20

Ramsey did not explicitly state SLED representation as input to generate a simulator tool. Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph) and that nML generates simulators (page 520, third line). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as simulator generation) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (simulation) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

7. Claims 16-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ramsey et al., "Specifying Representations of Machine Instructions" in view of Gupta et al. (USPN 6,385,757).

Claim 16

Ramsey disclosed the limitations:

i) *method for producing code in an architecture description language* (page 496, bracket 3)

ii) *reading an opcode summary table* (page 496, bracket A; page 497, bracket 5 indicates the opcode tables being read for information)

iii) *analyzing said opcode summary table to determine the layout of said opcode summary table* (page 495, bracket B illustrates differing types of instructions which would need to be analyzed in the table in order to be effectively implemented; page 495, bracket 1 indicates instructions differ and therefore to be properly handled the opcode table would need to be analyzed; page 497, bracket 5 indicates organizing the tables in a hierarchy for analysis purposes; page 499, bracket 9 indicates groups)

iv) *generating code for an instruction in architecture description language format* (page 495, section 2)

v) *repeating said generating step for each line on said opcode summary table* (necessary in order to build a complete machine description)

vi) *resulting in an architecture description language (ADL) representation of the opcode summary table* (Ramsey's resulting representation is in SLED an ADL) Ramsey did not explicitly state the method being computerized or in other words automated without a programmer. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey did not explicitly state SLED representation as input to generate a simulator tool. Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph) and that nML generates simulators (page 520, third line). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as simulator generation) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (simulation) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

Claim 17

Ramsey disclosed the limitations:

i) *method of producing code in an architecture description language format* (page 496, bracket 3)

ii) *reading an opcode summary table* (page 496, bracket A; page 497, bracket 5 indicates the opcode table being composed of several tables, but not any less a table)

iii) *analyzing said opcode summary table to determine the layout of said opcode summary table* (page 495, bracket B illustrates differing types of instructions which would need to be analyzed in the table in order to be effectively implemented; page 495, bracket 1 indicates instructions differ and therefore to be properly handled the opcode table would need to be analyzed; page 497, bracket 5 indicates organizing the tables in a hierarchy for analysis purposes)

iv) *determining the beginning of a group from said opcode summary table* (page 497, bracket 5; page 499, bracket 9)

v) *generating root code for the hierarchy in architecture description language format based on said grouping* (page 499, bracket 9; implicit patterns; page 497, bracket 6),

vi) *cycling through each group to generate detailed code in architecture language format* (necessary in order to build a complete machine description)

vii) *repeating said cycling step until the end of the opcode summary table is reached* (necessary in order to build a complete machine description)

Ramsey did not explicitly state the method being computerized or in other words automated without a programmer. Gupta demonstrated that it was known at the time of invention to "computerize" a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state SLED representation as input to generate a simulator tool. Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph) and that nML generates simulators (page 520, third line). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as simulator generation) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (simulation) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

Claim 18

Ramsey disclosed the limitations:

> i) *a first computer code section for reading an opcode summary table having a plurality of entries representative of a like plurality of microprocessor instructions* (page 496, bracket 3)
>
> ii) *a second computer code section for producing a group of at least two of said entries in accordance with a grouping criteria* (page 497, bracket 5; page 499, bracket 9)
>
> iii) *a third computer code section for generating an encoded representation of said grouping* (page 499, bracket 9)

Ramsey did not explicitly state the above steps being taken by a computer program. Gupta demonstrated that it was known at the time of invention to “computerize” a method of reading opcode tables to produce a description language (Gupta: column 4, line 65 to column 5, line 27; column 3, lines 49-54). It would have been obvious to one of ordinary skill in the art at the time of invention to implement Ramsey's ADL with the automated/computerized reading of the opcode table to produce the description language code as found in Gupta's teaching. This implementation would have been obvious because one of ordinary skill in the art would be motivated to produce automated descriptions to reduce design time (Gupta: column 3, lines 32-37).

Ramsey and Gupta did not explicitly state SLED representation as input to generate a simulator tool. Ramsey demonstrated that it was known at the time of invention to modify the nML ADL in accordance with SLED and conversely to modify SLED with features of nML (page 520, first full paragraph) and that nML generates simulators (page 520, third line). It would have been obvious to one of ordinary skill in the art at the time of invention to implement a system such as that described by Ramsey (for generating system dependent applications) with ADLs implemented with features of SLED (such as opcode summary tables) and other ADL features (such as simulator generation) as suggested by the above teachings. This implementation would have been obvious because one of ordinary skill in the art would be motivated to make use of varying benefits of the features of ADLs (simulation) and the benefits of opcode table features of SLED (Ramsey: page 493, paragraph 4; page 494, second full paragraph; page 495, first sentence; essentially easy to access specification).

Response to Arguments

8. Applicant's arguments filed 04 February 2004 have been fully considered but they are not persuasive. First, Applicant states (Response received 04 February 2004, page 9, third paragraph) that Examiner indicated SLED is not an ADL. This is untrue. Second, Applicant makes a case for use of *representation of structure*. As indicated above, this phrase does not have support in the specification. Third, Applicant argues for the new claims and limitations. These are addressed under the above rejections.

Applicant's arguments with respect to claims 1-20 have been considered but are moot in view of the new ground(s) of rejection.

Correspondence Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to William H. Wood whose telephone number is (703)305-3305. The examiner can normally be reached 7:30am - 5:00pm Monday thru Thursday and 7:30am - 4:00pm every other Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (703)305-9662. The fax phone numbers for the organization where this application or proceeding is assigned are (703)746-7239 for regular communications and (703)746-7238 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703)305-3900.

William H. Wood
April 12, 2004

TODD INGBERG
PRIMARY EXAMINER